CLAIMS

What is claimed is:

1. A method, comprising:

interconnecting a compute node with a shared memory node;

translating a processor instruction into an interconnect command;

transforming the interconnect command into a direct memory access interconnect command;

transmitting the direct memory access interconnect command via a link medium; and

performing an operation defined by the direct memory access interconnect command.

2. The method of claim 1, wherein the shared memory node is not immediately adjacent to the compute node.

3. The method of claim 1, wherein translating the processor instruction into the interconnect command includes translating a processor load instruction into an interconnect read command.

4. The method of claim 1, wherein transforming the interconnect command into the direct memory access interconnect command includes transforming the interconnect read command into a direct memory access interconnect read command.

5. The method of claim 1, wherein translating the processor instruction into the interconnect command includes translating a processor store instruction into an interconnect write command.

6. The method of claim 1, wherein transforming the interconnect command into the direct memory access interconnect command includes transforming the interconnect write command into a direct memory access interconnect write command.

7. The method of claim 1, wherein the link medium includes a serial link medium.

8. The method of claim 1, wherein the interconnect command defines a format.

9. The method of claim 8, wherein the format includes a local area network protocol.

10. The method of claim 8, wherein the format includes at least one field selected from the group consisting of a preamble, an address, a tag, a data, a cyclic redundancy check, and an end.

11. The method of claim 8, wherein the format includes a convolutional error detecting and correcting code.

12. The method of claim 10, wherein the tag contains a read command.

13. The method of claim 10, wherein the tag contains a write command.

14. The method of claim 10, wherein the address contains a specific location of shared memory in the shared memory node to be read.

15. The method of claim 10, wherein the address contains a specific location of shared memory in the shared memory node to be written.

16. The method of claim 5, further comprising:
receiving a write interconnect command from a host;
buffering a data;
reporting to the host that write interconnect command has been performed;
holding the data for a period of time; and
requesting that the host retry the write command at a later time.

17. The method of claim 5, further comprising combining a plurality of interconnect write commands into a single direct memory access interconnect write command of a plurality of data elements.

18. The method of claim 1, wherein the shared memory node modifies a memory address statically.

19. The method of claim 4, further comprising prefetching data with the compute node after translating the processor load instruction into the direct memory access interconnect read command.

20. The method of claim 19, wherein prefecthing includes:
recording characteristics of a series of load instructions;
analyzing characteristics of the series of load instructions;
determining a pattern in the series of load instructions; and
speculatively issuing direct memory access interconnect read commands to the shared memory node as a function of the pattern.

21. The method of claim 1, wherein interconnecting includes utilizing at least one member selected from the group consisting of: Peripheral Component Interconnect, Industry Standard Architecture, Small Computer System Interface, Universal Serial Bus, IEEE 1394, Micro Channel Architecture, and Extended Industry Standard Architecture.

22. An apparatus, comprising:
a computer network, including:
a compute node, having:
a compute node interconnect interface unit; and
a compute node interconnect adapter;
a link medium, coupled to the compute node; and

a shared memory node, coupled to the link medium, having:

a shared memory node interconnect interface unit; and

a shared memory node interconnect adapter.

23. The apparatus of claim 22, wherein the compute node interconnect interface unit translates a processor instruction into an interconnect command.

24. The apparatus of claim 22, wherein the compute node interconnect adapter transforms the interconnect command into a direct memory access interconnect command.

25. The apparatus of claim 22, wherein the compute node interconnect adapter is integrated with the compute node interconnect interface unit.

26. The apparatus of claim 22, wherein the shared memory node interconnect adapter is integrated with the shared memory node interconnect interface unit.

27. The apparatus of claim 22, wherein the compute node interconnect interface unit includes:

a processor bus;

a memory bus;

an interconnect bus;

a memory command translator; and

a memory address translator.

28. The apparatus of claim 22, wherein the shared memory interconnect interface unit includes:

a processor bus;

a memory bus;

an interconnect bus;

a memory command translator; and

a memory address translator.

29. The apparatus of claim 27, wherein the memory address translator includes a table of memory address ranges;

30. The apparatus of claim 29, wherein, wherein the table of memory address ranges includes a plurality of ranges associated with the interconnect bus.

31. The apparatus of claim 29, wherein the table of memory address ranges includes a plurality of ranges associated with the memory bus.

32. The apparatus of claim 29, wherein the table of memory address ranges is dynamically adjusted, with ranges determined at initialization time via a standardized test.

33. The apparatus of claim 32, wherein the standardized test includes reads and writes to certain required memory ranges within the compute node interconnect adapter.

34. The apparatus of claim 28, wherein the memory address translator includes a table of memory address ranges;

35. The apparatus of claim 34, wherein, wherein the table of memory address ranges includes a plurality of ranges associated with the interconnect bus.

36. The apparatus of claim 34, wherein the table of memory address ranges includes a plurality of ranges associated with the memory bus.

37. The apparatus of claim 34, wherein the table of memory address ranges is dynamically adjusted, with ranges determined at initialization time via a standardized test.

38. The apparatus of claim 37, wherein the standardized test includes reads and writes to certain required memory ranges within the shared memory node interconnect adapter.

39. The apparatus of claim 22, wherein the compute node interface adapter includes:

an interconnect bus interface;

an address translator;

a speculative-read control register;

a DMA-read control register;

a link protocol generator;

a link protocol responder;

a receive buffer;

a speculative-and-DMA read control calculator; and

a speculative-read control exerciser.

40. The apparatus of claim 22, wherein the shared memory node interconnect adapter, includes:

an interconnect bus interface;

an address translator;

an interconnect read/write state machine;

a link protocol generator; and

a link protocol responder.

41. The method of claim 20, wherein prefetching activation is performed via a link protocol responder.

42. The method of claim 41, wherein activation via the speculative read control register includes activation for a particular region of the shared memory node.

43. A method, comprising direct memory access by a shared memory node interconnect adapter to a shared memory.

44. An apparatus for performing the method of claim 1.

45. A computer program, comprising computer or machine readable program elements translatable for implementing the method of claim 1.

46. A hardware abstraction layer software, for implementing the method of claim 1.